ROUTES AVEC HORAIRES TRAITÉES PAR GÉNÉRATION DE COLONNES

par

Jacques Desrosiers**, François Soumis**
et Martin Desrochers**

Cette recherche a été rendue possible grâce à une subvention des Fonds internes de l'Ecole des H.E.C. et à deux subventions du C.R.S.N.G.

Cette recherche a également été publiée en anglais sous le titre de "Routing with Time Windows by Column Generation".

** Service des méthodes quantitatives, Ecole des Hautes Etudes Commerciales, 5255, av. Decelles, Montréal, Québec, Canada H3T 1V6.

Université de Montréal

Centre de recherche sur les transports - Publication #277
Ecole des Hautes Etudes Commerciales - Publication #G82-12

Mai 1983

Dépôt légal - 3e trimestre 1983

Résumé

Le problème considère un ensemble de parcours ayant chacun un lieu de début, un lieu de fin, une durée, un coût et un intervalle de temps pendant lequel ils doivent être effectués. Les parcours sont donnés a priori et peuvent consister en la visite d'un ou plusieurs points. Il faut trouver le nombre, l'horaire et les routes de véhicules pour effectuer chacun des parcours dans l'intervalle de temps imposé et ce en minimisant les coûts de déplacements entre les parcours. Ce problème est une généralisation du m voyageur de commerce.

Nous avons mis au point une méthode optimale traitant le temps comme une variable continue et trouvant simultanément les horaires et les routes de véhicules. La méthode procède par génération de colonnes produites par un sous-problème de chemin le plus court avec contraintes de temps. Des résultats numériques sont présentés.

1. INTRODUCTION

Ce travail présente une méthode pour traiter de façon exacte le problème de la fabrication des routes et des horaires d'une flotte de véhicules. Avant d'énoncer le problème, précisons d'abord la terminologie. Un parcours est un déplacement productif que devra effectuer un véhicule. Le parcours i est caractérisé par un lieu de début, un lieu de fin, une durée, un coût et un intervalle de temps $[a_i, b_i]$ pendant lequel i doit débuter. Un interparcours est un déplacement à vide que peut effectuer un véhicule. L'interparcours (i,j) va de la fin du parcours i au début du parcours j. Sa durée t_{ij} et sont coût c_{ij} peuvent inclure respectivement la durée et le coût du parcours i. Une route est une suite de parcours et d'interparcours effectués par un même véhicule.

Le problème consiste à déterminer les routes et l'horaire de façon à effectuer tous les parcours en minimisant le nombre de véhicules et les coûts de déplacements, en respectant les contraintes de réseau et les contraintes d'horaires.

Ce problème de fabrication de routes a été rencontré par les auteurs dans le cadre de plusieurs applications; autobus d'écoliers [6], autobus urbains [15], trains de voyageurs [14], flotte d'avions [16]. Ce problème se pose aussi lors de l'assignation à m machines de n tâches devant être effectuées chacune dans un intervalle de temps. Il a été traité dans ce contexte par Bratley, Florian, Robillard [3] et Gertsbakh, Stern [10].

1.1 Description des problèmes traités

L'algorithme a été évalué en traitant 6 problèmes de transport scolaire. Chaque problème est identifié par son nombre de parcours suivi de AM ou PM indiquant s'il s'agit du transport du matin ou de l'après-midi. Les problèmes AM et PM de même taille comprennent les mêmes parcours avec des intervalles de temps de même largeur mais la répartition dans le temps de ces intervalles est différente. Les fins de classe sont réparties sur une plus longue période que les débuts de classe comme l'indique la durée totale au tableau 1. Le nombre d'interparcours qui sont possibles est donc plus grand

l'après-midi et les problèmes sont très différents.

Tableau 1: Description des problèmes

Nombre de parcours	Nombre d'écoles	Durée des parcours			Durée $(b_i - a_i)$			Durée interparcours			Nombre interparcours		Durée totale	
		Min.	Max.	Moy.	% de 0	Max.	Moy.	Min.	Max.	Moy.	AM	PM	AM	PM
48	6	8	33	18.7	15%	9	4.6	1	29	10.9	579	620	80	110
100	46	13	76	34.4	34%	22	7.1	1	56	19.0	1137	2088	119	166
151	55	8	76	32.9	31%	22	7.8	1	56	18.5	2620	4900	121	166

En plus des 6 problèmes originaux, 40 variantes ont été traitées. 10 variantes de même difficulté que 48 AM et 48 PM ont été obtenues en perturbant aléatoirement l'horaire des écoles. 4 variantes plus difficiles de chacun des 6 problèmes originaux ont été construites en remplaçant les intervalles $[a_i, b_i]$ par $[a_i, b_i + \Delta]$ avec des valeurs de Δ de 5, 10, 15 et 20 minutes. Chaque nouveau problème de ce type est identifié par le nom du problème original suivi de $+\Delta$ (exemple 48 AM + 20). Les 6 variantes avec horaire fixe où le parcours i doit débuter à l'heure a_i, ont été traitées pour comparaison et sont identifiées en ajoutant au nom l'identification $t_i = a_i$ (par exemple, 48 AM $t_i = a_i$).

1.2 Formulation mathématique

Soit P l'ensemble des parcours et I l'ensemble des interparcours considérés. Pour qu'un interparcours soit considéré, il faut que l'on puisse effectuer le parcours j après le parcours i en respectant les intervalles de temps (c'est-à-dire $a_i + t_{ij} \leq b_j$). De plus l'ensemble I est généralement restreint par d'autres contraintes a priori sur les déplacements à vide.

Ce texte présente le problème avec un seul dépôt d'où chaque véhicule ne sort qu'une seule fois; les noeuds s et t représentent respectivement

la sortie et l'entrée au dépôt. Le réseau utilisé par les véhicules est défini par l'ensemble des noeuds $N = P \cup \{s,t\}$ et l'ensemble des arcs orientés $A = I \cup \{s\}\times P \cup P\times\{t\}$. D'autres variantes avec plusieurs dépôts ou avec plusieurs sorties par véhicules peuvent être traitées.

Deux types de variables sont utilisées: des variables de flot x_{ij}, $(i,j) \in A$ prenant la valeur 1 quand l'arc (i,j) est utilisé par un véhicule, et des variables continues de temps t_i, $i \in P$, associées à l'heure de début de chacun des parcours. Les coûts variables de déplacement sont associés aux variables de flots correspondant à des arcs d'interparcours, et le coût fixe par véhicule est associé aux variables de flot correspondant aux arcs de l'ensemble $\{s\}\times P$.

Les routes optimales respectant les contraintes d'horaires sont solution du problème suivant:

$$\text{Min} \sum_{(i,j)\in A} c_{ij}\, x_{ij} \qquad (1)$$

$$\sum_{j\in N} x_{ij} = \sum_{j\in N} x_{ji} = 1 \qquad i \in P \qquad (2)$$

$$x_{ij} \geq 0 \qquad (i,j) \in A \qquad (3)$$

$$x_{ij} > 0 \Rightarrow t_i + t_{ij} \leq t_j \qquad (i,j) \in I \qquad (4)$$

$$a_i \leq t_i \leq b_i \qquad i \in P \qquad (5)$$

$$x_{ij} \text{ binaire} \qquad (i,j) \in A \qquad (6)$$

Les relations (1), (2) et (3) forment un problème de fabrication de routes sans contrainte d'horaires. C'est un problème de flot à coût minimum qui se résoud économiquement et dont la solution est entière. Les relations (4) décrivent les exigences de compatibilité entre les routes et l'horaire. Les contraintes (5) imposent les intervalles de temps.

1.3 Propriétés

La proposition suivante montre que les contraintes (6) peuvent être supprimées de la formulation du problème.

Proposition 1

Il existe une solution optimale entière au problème de routes avec horaire défini par les relations (1) à (5).

Preuve: le problème (1)-(5) est réalisable; il suffit de prendre $(\forall i)$ $(x_{si} = 1,\ x_{it} = 1,\ t_i = a_i)$. Soit (X^*, T^*) une solution optimale. Si X^* n'est pas entier, définissons $J = \{(i,j) \mid x^*_{ij} > 0\}$. Maintenons T^* fixé et considérons le problème de flot à coût minimum sur le sous-réseau obtenu en conservant seulement les arcs de l'ensemble J. C'est un problème de flot pour lequel en plus de la solution optimale X^*, il existe une solution optimale entière $\bar{X}$. On peut facilement vérifier que $(\bar{X}, T^*)$ est une solution optimale entière de (1)-(5). □

Remarquons toutefois que même avec la proposition 1, la solution optimale entière ne peut être obtenue directement par programmation linéaire car les relations (4) ne sont pas des contraintes linéaires. Ces relations (4) peuvent s'écrire sous la forme linéaire suivante:

$$t_i + t_{ij} - t_j \leq (1 - x_{ij})\, M_{ij} \qquad (i,j) \in I \qquad (7)$$

avec $M_{ij} \geq b_i + t_{ij} - a_j$. Cette formulation est équivalente seulement si on impose x_{ij} binaire. Sinon pour $0 < x_{ij} < 1$, on peut avoir $0 < t_i + t_{ij} + t_j \leq (1 - x_{ij})\, M_{ij}$ qui ne satisfait pas (4). De plus, on peut facilement construire un exemple n'ayant pas de solution optimale entière quand on remplace (4) et (6) par (7).

Dans le cas particulier où $[a_i, b_i] = [0, |P|-1]$ pour $i \in P$ et $t_{ij} = 1$ pour $(i,j) \in I$, on peut poser $M_{ij} = |P|$ et les contraintes (7) deviennent:

$$t_i - t_j + |P|x_{ij} \leq |P|-1 \qquad (i,j) \in I. \qquad (8)$$

Il s'agit des contraintes d'élimination de sous-tours proposées par Miller, Tucker et Zemlin [12] pour le problème du voyageur de commerce. Dans ce cas particulier, notre problème devient celui du voyageur de commerce, alors qu'en général c'est un m-voyageur de commerce avec fenêtres de temps, les contraintes d'horaire assurant l'élimination des sous-tours.

2. METHODES DE RESOLUTION

Le problème formé des relations (1) à (5) est non-linéaire et difficile à résoudre. Les applications de grande taille ont jusqu'ici été traitées par des méthodes heuristiques. Certaines idées ont été proposées pour traiter ce type de problème de façon exacte.

2.1 Méthodes heuristiques

Une méthode heuristique de résolution discrétise les intervalles de temps et remplace chaque variable t_i par un ensemble de variables binaires associées à chacune des décisions de débuter ou non le parcours i à des points de la discrétisation. Levin [11] l'a utilisé en transport aérien et Swersey et Ballard [17] l'ont utilisé en transport scolaire; ils obtiennent fréquemment une solution optimale entière en résolvant avec l'algorithme du simplexe le problème sans contrainte d'intégralité. Remarquons toutefois qu'ils ont restreint l'objectif au nombre de véhicules après avoir constaté qu'il était plus difficile d'obtenir l'intégrité en considérant aussi les coûts de déplacements. D'autres méthodes présentées par Bodin et al [2] permettent de traiter aussi les coûts de déplacement mais elles utilisent des approximations supplémentaires.

2.2 Méthodes exactes

Une première approche exacte a été présentée par Orloff [13]. Il propose de reformuler ce problème comme un problème de recouvrement en énumérant explicitement toutes les routes qui satisfont les contraintes d'horaires. Toutefois Orloff remarque que le nombre de variables est très grand même pour un problème de taille moyenne et que cette approche ne peut donc être appliquée avec succès aux problèmes réels.

Les approches par branch-and-bound ont été les plus utilisées pour résoudre de façon optimale ce type de problème. On résoud un problème relaxé et effectue des branchements pour éliminer les solutions non-réalisables. Plusieurs relaxations ont été proposées; elles correspondent à des compromis différents entre le coût de résolution et la qualité de l'approximation. Pour le voyageur de commerce, Carpaneto et Toth [5] ont relaxé (4) et (5) pour traiter le problème d'affectation qui se résoud facilement. Cette méthode a été adaptée pour le problème plus général que nous traitons; quand (4) et (5) sont très contraignantes, l'écart entre le problème d'affectation et le problème original est important et l'arbre à explorer par le branch-and-bound est très grand. Pour diminuer cet écart et réduire la taille de l'arbre, Fisher [8] a utilisé la relaxation lagrangienne en incluant (4) et (5) dans l'objectif. Christofides, Mingozzi, Toth [14] ont proposé le "State-Space relaxation" qui utilise des relaxations intermédiaires approchant le problème original en partant de plus courts chemins ou d'arbres de recouvrement.

La relaxation en nombres réels obtenue en remplaçant (4) par son équivalent linéaire (6)-(7) et en relaxant l'intégralité (6) présente un écart plus petit que les relaxations précédentes. Ce résultat s'obtient à partir de l'"Integrality Property" du problème relaxé en utilisant les théorèmes 1 et 2 de Geoffrion [9]. Elle n'a pas été utilisée car la résolution avec l'algorithme du Simplexe d'un problème de $3*|P| + |I| = O(|P|^2)$ contraintes est trop coûteuxe. Notre approche utilise cette relaxation en résolvant le problème linéaire par génération de colonnes (Dantzig-Wolfe) pour profiter de la structure de réseau du problème. Le problème maître à traiter avec le Simplexe n'a que $|P|$ contraintes et le sous-problème se traite avec une adaptation de l'algorithme de plus court chemin.

3. RESOLUTION DE LA RELAXATION EN NOMBRES REELS

La méthodes de résolution est présentée en utilisant des variables de flot sur les routes et non sur les parcours. Une route satisfaisant les contraintes d'horaires (4) et (5) est caractérisée par la suite de parcours effectués et par l'heure de début de ces parcours. La reformulation du problème utilise la notation suivante:

Ω : ensemble des routes avec horaire satisfaisant les contraintes (4) et (5).

y_r : variable binaire: 1 si la route r est utilisée; 0 sinon.

δ_{ri} : constante binaire: 1 si le parcours i est effectué dans la route r; 0 sinon.

c_r : coût de la route r obtenu en sommant les coûts des arcs qui la composent: il comprend le coût fixe par véhicule et le coût de déplacement entre les parcours.

La résolution consiste à fabriquer un ensemble de routes de coût minimum satisfaisant les contraintes d'horaires et effectuant une fois chaque parcours. Ce problème de recouvrement se formule de la façon suivante:

$$\text{Min} \quad \sum_{r \in \Omega} c_r y_r \qquad (9)$$

$$\sum_{r \in \Omega} \delta_{ri} y_r = 1 \qquad i \in P \qquad (10)$$

$$y_r \text{ binaire.} \qquad (11)$$

A cause du grand nombre de routes dans Ω, le problème de recouvrement est résolu par génération de colonnes. Une nouvelle colonne de coût marginal minimum est générée en résolvant le problème (12)-(17) de plus court chemin avec contraintes d'horaires défini en utilisant les variables duales σ_i du problème de recouvrement:

$$\text{Min} \quad \sum_{(i,j)\in A} (c_{ij} - \sigma_i)\, x_{ij} \qquad (12)$$

$$\sum_{j\in N} x_{ij} = \sum_{j\in N} x_{ji} \qquad i \in P \qquad (13)$$

$$\sum_{j\in P} x_{sj} = \sum_{j\in P} x_{jt} = 1 \qquad (14)$$

$$x_{ij} > 0 \Rightarrow t_i + t_{ij} \le t_j \qquad (i,j) \in I \qquad (15)$$

$$a_i \le t_i \le b_i \qquad i \in P \qquad (16)$$

x_{ij} binaire. (17)

On peut montrer en utilisant l'argument de la Proportion 1 que ce sous-problème a une solution binaire même en éliminant la contrainte (17). Sa solution est une route de coût minimum de s à t satisfaisant les contraintes d'horaires.

3.1 Décomposition de Dantzig-Wolfe

La méthode de résolution par génération de colonnes correspond à une décomposition de Dantzig-Wolfe où les contraintes d'horaires (4) et (5) sont placées dans le sous-problème. Les contraintes de réseau (2) sont séparées entre le sous-problème et le problème maître: la conservation de flot dans le sous-problème et l'obligation de visiter chaque parcours dans le problème maître ($\sum_{j\in N} x_{ij} = 1, \quad \forall i$)

Suivant cette décomposition, le sous-problème doit s'écrire sans la contrainte (14); ses solutions sont alors, nulles ou non-bornées et des rayons extrémaux sont transmis au problème maître. La solution du problème maître est dans le cône convexe engendré par ces rayons extrémaux et n'a pas à satisfaire la contrainte de combinaison convexe. C'est ce qui explique son absence dans la formulation du problème de recouvrement (9)-(10).

En ajoutant la contrainte (14) on obtient un sous-problème borné ayant un point extrémal correspondant à chaque rayon extrémal du sous-problème non-borné. Remarquons que si le sous-problème avait été borné avec la contrainte $x_{ij} \leq 1$, provenant de la relaxation de x_{ij} binaire, sa résolution n'aurait pas été un plus court chemin mais un problème combinatoire aussi difficile que le problème original, c'est-à-dire la circulation de plusieurs véhicules de s à t en utilisant des routes n'ayant pas d'inter-parcours commun et un coût total le plus négatif possible.

3.2 Résolution du sous-problème

On trouve la route de coût minimum satisfaisant les contraintes d'horaires avec une adaptation de l'algorithme de Ford-Bellman que nous appelons plus court chemin avec contraintes d'horaires (PCCCH).

Rappelons que l'algorithme de Ford-Bellman, pour résoudre le problème classique du plus court chemin, associe à chaque noeud une étiquette π_i qui est le coût d'un chemin de s à i. Les étiquettes π_i sont améliorées par programmation dynamique jusqu'à l'obtention des valeurs optimales; π_i est alors le coût de l'origine s au noeud i par la route de coût minimum. Cet algorithme compare les routes en considérant seulement leur coût et une seule étiquette est conservée à chaque noeud car la valeur minimale de π_i est la seule utilisée pour calculer les valeurs minimales des autres π_j en accord avec le principe d'optimalité suivant:

> "Si r est une route de coût minimum de s à j et si (i,j)
> est son dernier arc, alors la sous-route de r, de s à i
> est une route de coût minimum"

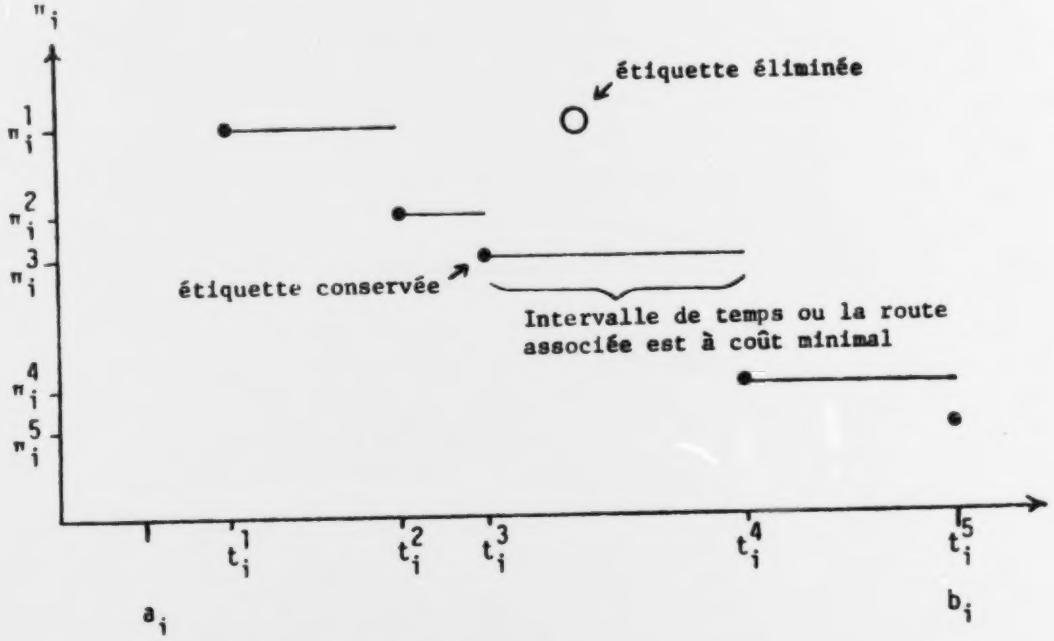
Pour le problème de la route de coût minimum avec contraintes d'horaires, les routes doivent être comparées en considérant les temps d'arrivée aux noeuds en plus des coûts. PCCCH utilise donc comme étiquettes au noeud i des couples (temps d'arrivée au noeud i, coût de la route). Plusieurs étiquettes doivent être conservées à un noeud i car elles peuvent être utilisées pour calculer les étiquettes des autres noeuds en accord avec le principe d'optimalité suivant:

"Si r est une route de coût minimum parmi les routes de s à j qui arrivent en j à l'heure t_j ou avant et si (i,j) est son dernier arc, alors la sous-route de r, de s à i est une route de coût minimum parmi les routes de s à i qui arrivent en i à l'heure $t_j - t_{ij}$ ou avant"

L'algorithme conserve au noeud i une étiquette (t_i, π_i) tant que l'on a pas trouvé une route moins coûteuse qui arrive en i au temps t_i ou avant.

L'ensemble des étiquettes conservées à un noeud $(t_i^\alpha, \pi_i^\alpha)$, $\alpha \geq 1$ se réduit à une liste finie Q_i que l'on classe par ordre décroissant des coûts. Remarquons, comme l'illustre la figure 1, que cette liste est en même temps en ordre croissant des temps. L'amélioration de la liste Q_j à partir de Q_i en considérant les routes de l'origine à j constituée des routes de l'origine à i et de l'arc (i,j) est une mise à jour séquentielle de Q_j à partir de l'information contenue dans Q_i. Une structure de données et une stratégie de gestion des listes d'étiquettes permettant le traitement efficace de ce problème ont été présentées dans Desrosiers, Pelletier, Soumis [7].

Figure 1: Coût vs heure d'arrivée des étiquettes conservées au noeud i

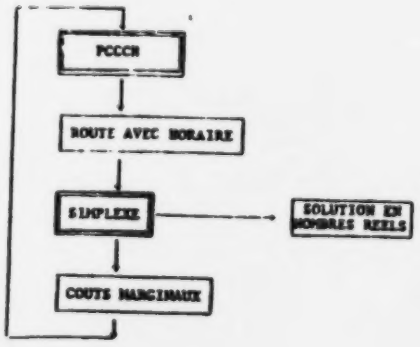


Cette méthode de résolution des sous-problème peut produire des routes avec cycles si le réseau comprend des cycles de coûts négatifs satisfaisant les contraintes d'horaires. De tels cycles sont rares quand les fenêtres de temps sont petites car pour qu'un tel cycle existe, il faut pouvoir au moins effectuer un premier parcours, un interparcours, un second parcours et un interparcours ramenant le véhicule au début du premier parcours pendant l'intervalle de temps où le premier parcours peut débuter. Si une route avec cycle est produite, elle est d'une durée et d'un coût fini à cause des intervalles de temps bornés. Elle est ajoutée au problème maître avec $\delta_{ri} > 1$ pour les parcours utilisés plus d'une fois. Une telle colonne peut entrer dans la base à un niveau fractionnaire mais redevient nulle par la suite par une itération de simplexe ou par branchement car elle ne peut prendre la valeur 1.

3.3 Résolution du problème maître

Le problème maître est traité avec l'algorithme du Simplexe. Cette méthode donne les coûts marginaux σ_i nécessaires pour la génération de colonnes et se prête bien à la réoptimisation chaque fois que de nouvelles colonnes sont générées. La figure 2 présente une itération majeure: la génération de routes avec horaire, suivie de la réoptimisation du problème de recouvrement. Ces itérations conduisent à la solution optimale en nombres réels.

Figure 2 : Génération de colonnes



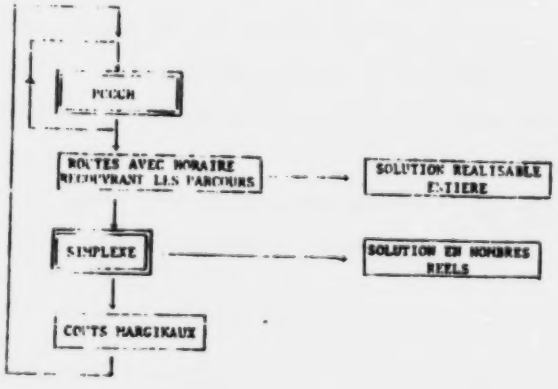
Les premiers essais ont montré que la plus grande partie du temps est utilisée dans le Simplexe tandis que PCCCH consomme peu de temps. La procédure fut modifiée pour accélérer la convergence en générant plusieurs routes par itérations majeures. Cette modification exploite l'idée qu'il est bon d'inclure dans l'ensemble des routes générées $\Omega' \subseteq \Omega$, toutes routes qui ont une probabilité d'être optimales et même des routes non optimales qui peuvent permettre au cours des itérations de réduire l'objectif et d'améliorer les variables duales. Cette idée permet de réduire le nombre d'itération du simplexe qui est assez coûteux en augmentant l'utilisation de PCCCH qui est peu coûteux.

Plusieurs stratégies de génération de multiples colonnes ont été essayées. La méthode suivante s'est avérée supérieure (figure 3):

1° générer la route de coût minimum satisfaisant les contraintes d'horaires;

2° retirer du réseau les parcours utilisés par cette route et retourner en 1° s'il reste des parcours.

Figure 3 : Génération de multiples colonnes par itération



Cette méthode a l'avantage de fabriquer des routes recouvrant tous les parcours comparativement à d'autres méthodes qui fabriquent des routes n'incluant que les parcours peu coûteux. De plus, en fabriquant des routes disjointes, on favorise l'apparition de solutions entières. Le tableau 2 présente la comparaison entre les générations de colonnes simple et multiple pour les problèmes 48 AM, 48 PM et les 10 problèmes obtenus en modifiant aléatoirement l'horaire des écoles.

Tableau 2: Comparaison entre deux stratégies de génération de colonnes

	Nombre d'itérations majeures	Simplexe		Génération de colonnes		Temps total
		Nombre d'itérations	Temps (sec)	Nombre de colonnes	Temps (sec)	
1 colonne par itération	105	761	24.5	153	4.5	29.0
Plusieurs col. par itération	14	356	11.1	255	3.5	14.6

La seconde stratégie réduit sensiblement le nombre d'itérations majeures (réoptimisation du problème maître). Le nombre d'itérations et le temps pour le Simplexe sont réduits de moitié. Même si la seconde méthode génère plus de colonnes, elle le fait en moins de temps car en enlevant les parcours utilisés par la colonne générée, on diminue la taille du réseau pour le calcul des plus courts chemins suivants.

4. RESOLUTION EN NOMBRES ENTIERS

Pour ce problème, la probabilité d'obtenir spontanément une solution entière du problème maître avec le Simplexe est assez forte. L'apparation de solutions entières est favorisée en utilisant au besoin des coupes exigeant de compléter le nombre de véhicules et le coût de déplacement aux entiers suivants. Quand la solution n'est pas entière, il n'y a pas tellement de variables fractionnaires à éliminer. On utilise donc, pour trouver la solution optimale entière, une procédure d'énumération implicite avec le Simplexe pour résoudre le problème à chaque noeud de l'arbre.

4.1 Les coupes

L'expression (18) sépare l'objectif en deux termes: les coûts fixes associés au nombre de véhicules et le coût de déplacement

$$C^*\left(\sum_{j\in P} x_{sj}\right) + \sum_{(i,j)\in I} c_{ij}\, x_{ij} \qquad (18)$$

La somme des flots sur les arcs de sortie de s vers tous les parcours compte le nombre de véhicules utilisés, C étant le coût fixe par véhicule. Le second terme est la somme des coûts de déplacement sur les interparcours utilisés (le coût des parcours n'est pas inclus dans l'objectif). Pour les applications traitées, le coût fixe C est suffisamment grand pour qu'il ne soit pas rentable d'augmenter le nombre de véhicules pour réduire le coût de déplacement. Ainsi la solution optimale minimise le nombre de véhicules et pour ce nombre de véhicules, minimise le coût de déplacement.

Quand le nombre de véhicules n'est pas entier dans la solution optimale du problème maître, on ajoute une coupe exigeant de compléter le nombre de véhicules au moins à l'entier suivant. Le problème maître est alors réoptimisé en générant au besoin de nouvelles colonnes. L'efficacité de cette coupe est illustrée dans le tableau 3 qui présente les 14 exemples où le nombre de véhicules n'était pas entier parmi les 40 problèmes traités.

Tableau 3: Effet de la coupe exigeant de compléter le nombre de véhicules au moins à l'entier suivant.

Problème	Coupe sur véh. entiers	Solution réelle			1ère solution entière		Meilleure solution entière		Résultat final		
		Autobus	Coût	Temps (CDC 173)	Noeud	Coût	Noeud	Coût	Noeuds explorés	Temps (CDC 173)	Optimale
[illegible]	sans	13.3	--	38	11	232	20	229	56	502	non
	avec	14	209	65	1	209	1	209	1	65	oui
[illegible]	sans	12.4	--	58	7	245	54	212	57	453	non
	avec	13	205	89	6	207	6	207	14	427	non
[illegible]	sans	11.7	--	100	9	233	12	223	26	560	non
	avec	12	210	176	1	210	1	210	1	176	oui
[illegible]	sans	11.9	--	64	7	388	7	388	23	480	non
	avec	12	362.8	95	5	366	10	364	14	425	oui
[illegible]	sans	11.1	--	85	10	333	16	332	22	541	non
	avec	12	312.7	151	7	329	7	329	11	434	non
[illegible]	sans	8.5	--	38	9	408	9	408	32	348	non
	avec	9	394	64	1	394	1	394	1	64	oui
[illegible]	sans	54.5	--	48	5	480	5	480	50	368	non
	avec	55	475.5	79	7	480	7	480	160	426	oui
[illegible]	sans	50.5	--	55	13	593	19	586	30	156	non
	avec	51	571	118	1	571	1	571	1	118	non
[illegible]	sans	43.5	--	118	7	647	14	639	38	437	non
	avec	44	635	116	3	635	3	635	3	118	oui
[illegible]	sans	39.5	--	157	9	708	9	708	13	468	non
	avec	40	688.5	206	3	692	3	692	9	404	non
[illegible]	sans	87.3	--	170	28	618	28	618	43	689	non
	avec	88	595	183	2	595	2	595	2	185	oui
[illegible]	sans	73.5	--	251	17	810	24	808	32	978	non
	avec	74	781	401	3	781	3	781	3	511	oui
[illegible]	sans	69.5	--	358	--	--	--	--	15	1000	non
	avec	70	903	519	3	903	3	903	3	634	oui
[illegible]	sans	65.5	--	673	--	--	--	--	16	2000	non
	avec	66	943	692	1	943	1	943	1	690	oui

Pour chaque problème la première ligne présente les résultats obtenus sans la coupe, la seconde ligne ceux obtenus en ajoutant la coupe sur le nombre véhicules. Le coût de déplacement n'est pas donné quand le nombre d'autobus est fractionnaire car les fractions de véhicules et le coût de déplacement se confondent dans notre objectif. Pour la première et la meilleure solution entière, les colonnes "Noeud" indiquent les nombres de noeuds énumérés avant d'obtenir ces solutions. La colonne "Noeuds explorés" pour le résultat final indique le nombre total de noeuds explorés dans l'arbre d'énumération. Toutes les solutions entières obtenues avec ou sans la coupe sont optimales pour le nombre de véhicules; elles utilisent un nombre de véhicules égal à l'entier suivant le nombre fractionnaire donné par la solution réelle sans la coupe. L'algorithme avec cette coupe sur le nombre de véhicules obtient des

solutions optimales pour le coût de déplacement pour 10 problèmes et pour les autres l'écart entre la borne et la solution trouvée est de 2, 3, 4 et 16 minutes. L'algorithme sans coupe n'est optimal pour aucun problème et les temps de déplacements sont en moyenne supérieurs de 20 minutes.

Par la suite quand le coût de déplacement n'est pas entier, on ajoute une seconde coupe exigeant de le compléter au moins à l'entier suivant et on réoptimise. Le tableau 4 illustre l'effet de cette seconde coupe pour les 7 problèmes qui ont des coûts de déplacements fractionnaires après avoir inséré au besoin la coupe sur le nombre de véhicules.

Tableau 4: Effet de la coupe exigeant de compléter le coût de déplacement au moins à l'entier suivant

Problème	Coupe sur le déplacement	Solution réelle			1ère solution entière		Meilleure sol. entière		Résultat final		
		Autobus	Coût	Temps (CDC 173)	Noeud	Coût	Noeud	Coût	Noeuds explorés	Temps (CDC 173)	Optimale
48 PM+5	sans	14	380.6	33	6	389	30	387	64	450	?
	avec	14	381	31	12	388	15	387	50	711	?
48 PM+10	sans	13	370.2	52	6	373	6	373	44	450	?
	avec	13	371	61	3	373	3	373	30	576	?
48 PM+15	sans	12	362.8	95	5	366	10	364	14	425	oui
	avec	12	363	101	3	364	3	364	10	334	oui
48 PM+20	sans	12	312.7	151	7	329	7	329	11	434	non
	avec	12	313	161	9	318	9	318	16	755	?
100 AM+10	sans	55	475.5	79	7	480	7	480	140	426	non
	avec	55	476	83	2	480	11	479	153	827	?
100 PM+15	sans	40	688.5	173	3	692	3	692	9	404	non
	avec	40	689	177	3	690	3	690	76	897	?
1 1 PM	sans	75	850.1	283	5	860	18	854	24	1109	non
	avec	75	851	301	2	851	2	851	2	378	oui

Pour chaque problème, la première ligne présente les résultats obtenus en utilisant au besoin la coupe sur le nombre de véhicules, la seconde ligne présente les résultats obtenus en utilisant aussi la coupe sur le coût de déplacement. En utilisant la seconde coupe, on explore moins de noeuds avant d'obtenir la première solution entière et la solution obtenue est meilleure. La comparaison des meilleures solutions entières obtenues est aussi à l'avantage de la méthode avec deux coupes. Toutefois, même avec deux coupes, plusieurs de ces problèmes ne sont pas résolus à l'optimalité, mais les solutions obtenues sont très près de l'optimalité.

Les deux coupes peuvent être placées dès le départ dans le tableau du Simplexe car les coefficients des variables sont connus. Les membres droits sont d'abord fixés à zéro pour que les coupes soient inactives. Quand on veut imposer une de ces coupes, on augmente le membre droit à la valeur désirée; de même, on le diminue pour relaxer la coupe. C'est avantageux car la seconde coupe doit être resserrée à mesure que l'on descend dans l'arbre d'énumération en fixant des variables et relaxée quand on remonte explorer les les noeuds pendants.

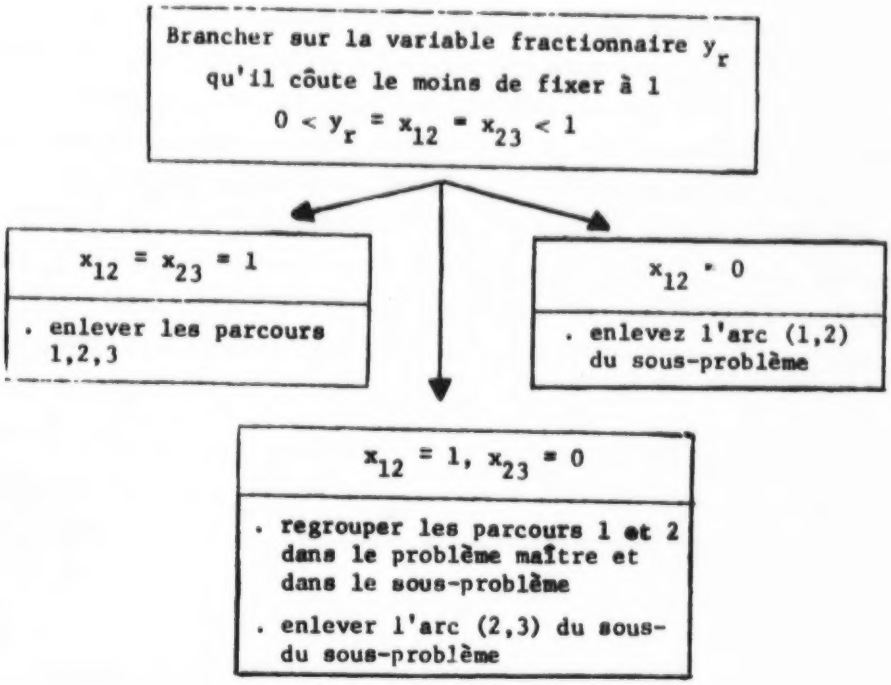
4.2 Arbre d'énumération

Le branchement à un noeud s'effectue sur la route avec flot fractionnaire pour laquelle il en coûterait le moins d'augmenter ce flot à un, selon l'estimation donnée par les coûts marginaux. La partition d'un noeud s'effectue suivant la méthode introduite par Bellmore et Malone [1] et utilisée par Carpaneto et Toth [5] pour le voyageur de commerce. Le branchement ne s'effectue pas en fixant les variables y_r mais en fixant à 0 ou à 1 les k variables x_{ij} associées aux arcs qui composent la route r. Les 2^k façons de fixer ces variables sont regroupées en k + 1 branches comprenant respectivement 1, 1, 2, 4,..., 2^{k-1} façons. La première branche fixe les k arcs à 1. La seconde fixe les k - 1 premiers arcs à 1 et le dernier à 0. La troisième fixe les k - 2 premiers arcs à 1 et le suivant à 0. Ainsi de suite jusqu'à la k + 1 ième qui fixe le premier arc à 0. Ce type de branchement permet de conserver dans chaque branche un problème maître et un sous-problème de même type que ceux du départ. Un branchement fixant $y_r = 0$ aurait détruit la structure du plus court chemin des sous-problèmes.

Le branchement proposé ne s'applique pas pour les routes d'un seul parcours car le nombre d'arc k est nul. De telles routes peuvent être présentes dans le problème comme variables d'écart et être fractionnaires. La difficulté est contournée en branchant uniquement sur les variables fractionnaires correspondant à des routes de deux parcours ou plus. Quand ces variables sont toutes entières, les variables d'écart le sont aussi. Pour faciliter la présentation des branches considérées et de leur traitement, considérons un exemple où le branchement s'effectue sur la route utilisant

dans l'ordre les parcours 1, 2 et 3. La figure 4 présente les branches considérées.

Figure 4: L'arbre d'énumération implicite



Dans l'arbre d'énumération implicite, on traite toujours le noeud le plus à gauche. Cette descente sur la gauche réduit la taille du problème et trouve rapidement une solution entière.

4.3 Traitement de chaque branche

Pour la branche gauche $x_{12} = x_{23} = 1$, la solution comprend la route constituée des parcours 1, 2 et 3. Pour trouver les autres routes, on enlève les lignes 1, 2 et 3 du problème maître et les routes contenant un de ces parcours. On enlève dans le sous-problème les noeuds correspondant à ces parcours. On réoptimise le problème de recouvrement en générant au besoin de nouvelles routes.

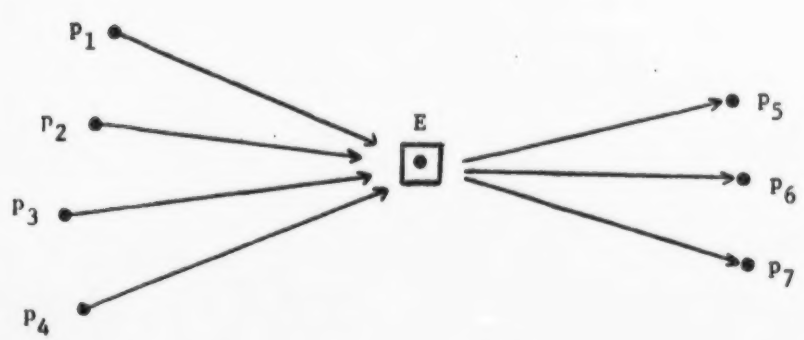
Pour la branche centrale $x_{12} = 1$, $x_{23} = 0$, la solution comprend une route utilisant les parcours 1 et 2 mais pas le parcours 3. Cette route peut être complétée en y ajoutant d'autres parcours. Pour compléter cette route et trouver les autres routes constituant une solution, on fait l'agrégation des lignes 1 et 2 du problème maître que l'on appelle ligne 1∧2. Parmi les routes contenant le parcours 1 ou le parcours 2, on ne conserve que celles contenant à la fois les parcours 1 et 2 et ne contenant pas le parcours 3. On leur attribue la valeur 1 sur la ligne 1∧2 du problème maître. De même, dans le sous-problème, on fait l'agrégation des parcours 1 et 2. On associe au nouveau parcours 1∧2 l'intervalle de temps $[a_1, \min\{b_1, b_2 - t_{12}\}]$. Les arcs qui arrivaient en 1 arrivent maintenant en 1∧2. Ceux qui arrivaient en 2 sont supprimés. Les arcs qui partent de 1 sont supprimés. L'arc (2,3) est supprimé. Les autres arcs qui partaient de 2 partent maintenant de 1∧2 avec des coûts et des durées augmentés respectivement de c_{12} et t_{12}.

Pour la branche de droite $x_{12} = 0$, la solution ne comprend pas de routes utilisant le parcours 1 suivi du parcours 2. On élimine de telles routes du problème maître et on enlève l'arc (1,2) du sous-problème pour éviter d'en générer d'autres.

5. AGREGATION DU RESEAU

Les interparcours constituent la majorité des arcs du réseau et leur nombre approche $|P|(|P|-1)/2$. Ce nombre a été réduit en utilisant des noeuds écoles tel qu'illustré à la figure 5.

Figure 5: Agrégation du réseau scolaire



L'exemple présente un problème de l'après-midi où p_5, p_6 et p_7 représentent tous les parcours sortant d'une école E, et où p_1 à p_4 sont les parcours pouvant terminer assez tôt pour qu'ils soient suivis d'un des parcours débutant à l'école E. On remplace les interparcours de $\{p_1, p_2, p_3, p_4\}$ vers $\{p_5, p_6, p_7\}$ par des arcs de $\{p_1, p_2, p_3, p_4\}$ vers E et des arcs de E vers $\{p_5, p_6, p_7\}$. En effectuant cette agrégation le nombre maximum d'arcs pour cette école devient $3 + 4 = 7$ arcs, tandis qu'il était $3 \times 4 = 12$ arcs dans le réseau initial, l'économie obtenue en remplaçant un produit par une somme est d'autant plus important que les nombres sont grands.

Remarquons que si les interparcours (1,5) et (1,6) existent tandis que l'interparcours (1,7) n'existe pas ($a_1 + t_{17} > b_7$) dans le réseau initial, l'agrégation ajoute un chemin de 1 vers 7. Toutefois le calcul des plus chemins avec contraintes d'horaires n'est pas affecté, car ce nouveau chemin arrive au parcours 7 après b_7 et n'est pas considéré. Remarquons aussi que pendant l'énumération implicite il n'est pas possible de fixer un interparcours à 0 dans le réseau agrégé. Ainsi pour fixer l'interparcours (1,5) à 0 on enlève les arcs (p_1, E) et (E, p_5) et on rétablit les interparcours de $\{p_1\}$ vers $\{p_6$ et $p_7\}$ et de $\{p_2, p_3, p_4\}$ vers $\{p_5\}$. L'économie de mémoire et le gain de temps dans PCCCH l'emportent de loin sur le temps de désagrégation pendant l'énumération implicite étant donné le peu de noeuds explorés. Le tableau 5 présente les nombres d'arcs avant et après agrégation pour les problèmes traités.

Tableau 5: Effet de l'agrégation sur la taille du réseau

Num du Problème	Problème original		Variantes du problème							
			$\Delta = 5$		$\Delta = 10$		$\Delta = 15$		$\Delta = 20$	
48 AM	579[1]	133[2]	669	145	764	160	876	177	999	193
48 PM	620	135	711	143	800	157	888	169	999	180
100 AM	1137	657	1532	814	1793	927	2093	1060	2505	1264
100 PM	2088	971	2478	1112	2797	1232	3018	1322	3338	1462
151 AM	2620	1114	3572	1392	4311	1635	5108	1899	6068	2240
151 PM	4900	1706	5787	1938	6529	2151	7077	2322	7795	2568

(1) NOMBRE D'ARCS AVANT L'AGREGATION

(2) NOMBRE D'ARCS APRES L'AGREGATION

6. EXPERIMENTATION

Les resultats sont divisés en deux sections; les résultats pour les 5 problèmes originaux et une étude de l'effet d'élargir les intervalles de temps $[a_i, b_i]$.

6.1 Les problèmes originaux

Ces problèmes ont été résolus facilement car les intervalles de temps sont assez étroits. Des problèmes plus grands auraient pu être traités si ce n'était que la taille limite de la mémoire était atteinte. Pour traiter de plus grands problèmes, il faudrait remplacer l'implantation du Simplexe de Land-Powell que nous utilisons par une autre implantation qui conserve l'inverse de la base sous forme produit plutôt que sous forme explicite.

Tableau 6 Résultats pour les 6 problèmes originaux

Problème	Colonnes générées	Etiquettes par parcours	Noeuds explorés	Temps de calcul	Solution optimale		Economie de véhicules
					Autobus	Coût	
48 AM	163	1.5	6	16/4.1	17	141	15%
48 PM	156	2.	1	13/3.2	17	320	29%
100 AM	238	1.6	1	12/14	63	390	10%
100 PM	290	2.1	1	35/24	51	571	14%
151 AM	393	2.	1	65/49	94	560	13%
151 PM	473	2.	2	307/69	75	851	18%

On remarque que tous ces problèmes ont une solution entière avec le même nombre d'autobus et le même coût que la borne trouvée par la relaxation en nombres réels avec les coupes. L'arbre d'énumération implicite contient peu de noeuds et sa taille n'augmente pas avec le nombre de parcours. Le nombre de colonnes du problème maître qui sont générées augmente linéairement avec le nombre de parcours. Le nombre moyen d'étiquettes générées à chaque parcours du réseau par l'algorithme du plus court chemin avec contrainte d'horaire est très petit et n'augmente pas avec la taille du réseau. Le temps de calcul en secondes sur CDC 173 (temps du simplexe/temps du plus court chemin)

n'augmente pas de façon exponentielle avec la taille du problème. La valeur discordante de 307 secondes pour 151 PM est due aux nombreuses réinversions qui ont été effectuées. La dernière colonne donne l'économie de véhicules par rapport à la flotte nécessaire pour effectuer ces parcours à horaire fixe. Remarquons l'importance des économies même si les intervalles $[a_i, b_i]$ sont assez étroits.

6.2 Effet de la largeur de l'intervalle $[a_i, b_i]$

Le tableau 7 présente la résolution de variantes des problèmes où les intervalles $[a_i, b_i]$ ont été graduellement agrandis jusqu'à une largeur de l'ordre de la durée des parcours dans les cas $\Delta = 20$. Ces variantes des problèmes sont beaucoup plus difficiles, le nombre d'arcs est à peu près doublé pour $\Delta = 20$ et on s'éloigne des problèmes de réseau cas $t_i = a_i$ comme l'indique les réductions jusqu'à 50% du nombre d'autobus utilisés. Les temps de calcul augmentent avec la largeur des intervalles mais la plupart des problèmes sont encore résolus à l'optimum. Pour les variantes où l'optimalité n'est pas prouvée, le nombre d'autobus est minimal, et l'écart sur le coût de déplacement est moins de 1.6%. Ce pourcentage est réduit à 0.5% si on considère le déplacement total des véhicules. Remarquons que les problèmes où l'optimalité n'a pas été prouvé ne sont ni les plus grands problèmes ni ceux ayant les intervalles $[a_i, b_i]$ les plus larges. Les performances de l'algorithme varient avec la largeur des intervalles de la façon suivante! Le nombre de noeuds explorés dans l'arbre d'énumération avant d'atteindre le meilleure solution entière n'augmente pratiquement pas. Le nombre de colonnes générées augmente à peu près comme le nombre d'arcs. Le nombre d'étiquettes par parcours, dans l'algorithme du plus court chemin avec contraintes d'horaires, augmente lentement. Le temps de calcul du Simplexe augmente plus vite que le temps de calcul des plus courts chemins.

Tableau 7: Effet de la largeur de l'intervalle $[a_i, b_i]$

Problème	Nombre d'arcs	Simplexe + coupes			Meilleure solution entière						Résultats finaux		
		Autobus	Coût	Temps calcul[1]	Autobus	Coût	Noeud[2]	Optimal	Econ.[3] de véhicules	Noeuds explorés[4]	Colonnes[5]	Arêtes[6]	Temps de calcul
50 AM $[a_i, b_i]$	490	20	185	1.1/1.6	20	185	1	oui	--	1	129	1	3.1/1.6
50 AM	579	17	191	5.3/3.1	17	191	6	oui	15%	6	163	1.5	16.5/4.1
50 AM+5	669	15	213	13/5.3	15	213	2	oui	25%	2	174	2	19.1/5.7
50 AM+10	766	14	209	23/15	14	209	2	oui	30%	2	262	5	94/18
50 AM+15	876	13	205	33/25	13	207	6	?	35%	26	424	9	375/43
50 AM+20	999	13	210	44/36	12	210	1	oui	40%	1	307	14	94/71
50 PM $[a_i, b_i]$	512	24	212	2.6/1.3	24	212	1	oui	--	1	121	1	2.6/1.3
50 PM	620	17	320	13/3.2	17	320	1	oui	29%	1	156	2	13/3.2
50 PM+5	711	14	381	25/7.8	14	387	15	?	42%	50	431	3.4	687/74
50 PM+10	800	13	371	86/16	13	373	3	?	46%	10	402	5.4	526/40
50 PM+15	888	12	363	85/29	12	364	3	oui	50%	10	365	7.8	274/60
50 PM+20	999	12	313	34/51	12	318	9	?	50%	16	404	11	668/87
100 AM $[a_i, b_i]$	787	70	286	5.5/4.6	70	286	1	oui	--	1	181	1	5.5/4.6
100 AM	1137	63	390	12/14	63	390	1	oui	10%	1	238	1.6	12/14
100 AM+5	1532	59	394	18/20	59	394	1	oui	11%	1	260	1.8	31/21
100 AM+10	1793	55	476	24/24	55	479	11	?	21%	153	420	2	756/71
100 AM+15	2093	51	515	53/36	51	515	2	oui	27%	2	299	2.5	67/36
100 AM+20	2505	46	608	59/57	46	608	3	oui	34%	3	337	3.5	80/63
100 PM $[a_i, b_i]$	1703	59	464	4.9/5.8	59	464	1	oui	--	1	175	1	4.9/5.8
100 PM	2088	51	571	33/22	51	571	1	oui	14%	1	290	2.1	35/24
100 PM+5	2478	46	644	41/32	46	644	1	oui	22%	1	308	2.1	41/32
100 PM+10	2797	44	635	74/44	44	635	2	oui	25%	2	347	2.6	185/47
100 PM+15	3018	40	689	115/55	40	690	3	?	37%	76	473	3	807/90
100 PM+20	3338	37	736	176/74	37	736	77	oui	37%	7	455	3.1	379/102
151 AM $[a_i, b_i]$	1535	108	406	11/10	108	406	1	oui	--	1	261	1	11/10
151 AM	2620	94	560	45/49	94	560	1	oui	13%	1	393	2	45/49
151 AM+5	3572	88	595	90/80	88	595	1	oui	19%	1	427	2.3	128/82
151 AM+10	4311	82	734	263/119	82	734	1	oui	24%	1	454	3	263/119
151 AM+15	5108	74	781	183/68	74	781	3	oui	31%	3	518	3.2	319/192
151 AM+20	6094		(mémoire > 200 000$_8$)										
151 PM $[a_i, b_i]$	3749	92	654	14/18	92	654	1	oui	--	1	298	1	14/18
151 PM	4900	75	850.1	215/67	75	851	2	oui	18%	2	473	2	907/69
151 PM+5	5787	70	903	354/104	70	903	6	oui	24%	6	525	2.3	648/122
151 PM+10	6529	66	943	437/133	66	943	1	oui	28%	1	546	3.5	577/135
151 PM+15	7077	58	1059	633/261	58	1059	4	oui	37%	4	637	4.2	437/269
151 PM+20	7795		(mémoire > 200 000$_8$)										

1- TEMPS DE CALCUL SUR CDC 173, SECONDES POUR SIMPLEXE/SECONDES POUR PCCCH
2- NOMBRE DE NOEUDS ENUMERES AVANT D'ATTEINDRE LA MEILLEURE SOLUTION
3- ECONOMIE DE VEHICULES EN COMPARAISON AVEC DES PARCOURS A HEURE-FIXE
4- NOMBRE TOTAL DE NOEUDS ENUMERES
5- NOMBRE TOTAL DE COLONNES GENEREES
6- NOMBRE MOYEN D'ETIQUETTES PAR PARCOURS DANS PCCCH

7. CONCLUSION

En traitant les contraintes d'horaires lors de la fabrication des routes avec l'algorithme PCCCH et en résolvant un problème de recouvrement avec le Simplexe, on perd l'intégrité de la solution mais on obtient une excellente borne inférieure. Cette borne ne se détériore pas avec l'augmentation du nombre de parcours ou l'élargissement des fenêtres de temps. Elle donne le nombre optimal de véhicules pour tous les problèmes traités et le temps de parcours minimal 4 fois sur 5; dans les autres cas l'écart est de l'ordre de 1%. Il est facile de retrouver l'intégrité par la suite, le simplexe donne souvent la solution entière directement, ou on l'obtient en réoptimisant avec le simplexe après fixation de quelques variables. Dans les quelques cas où cette première solution entière a un coût supérieur à la borne du simplexe, le branchement de Dantzig et Malone est peu efficace pour améliorer la solution ou prouver l'optimalité. L'arbre à énumérer est très grand même si l'écart est très petit.

L'algorithme n'a pas été utilisé pour des problèmes de plus de 151 parcours à cause de la limite de $200\ 000_8$ mots-mémoire d'ordinateur disponible. Des problèmes plus grands pourraient être traités car les temps de calcul n'augmentent pas exponentiellement avec le nombre de parcours. Ce traitement serait possible en utilisant un autre ordinateur ou un algorithme du Simplexe qui conserve l'inverse de la base sous forme produit.

Pour les problèmes où les fenêtres de temps sont relativement larges on a une réduction jusqu'à 50% de la flotte nécessaire pour effectuer les mêmes parcours à horaire fixe. Il n'est pas efficace d'utiliser pour ces problèmes des méthodes [5] traitant d'abord un problème de réseaux correspondant à un horaire fixe puis de modifier ensuite le réseau dans un branch and-bound pour obtenir une solution satisfaisant les contraintes d'horaires. Même si l'effort à chaque noeud du branch-and-bound est moindre et que les solutions obtenues sont entières, on ne peut résoudre que de petits problèmes. Les premières solutions sont loin de l'optimum et l'arbre à énumérer pour les améliorer croît rapidement avec la taille du problème et la largeur des intervalles de temps.

L'algorithme développé ouvre la possibilité de réduire la flotte de véhicules nécessaire pour effectuer un ensemble de parcours. Il peut traiter de grand problèmes et des fenêtres assez larges pour réaliser des économies substantielles.

BIBLIOGRAPHIE

[1] Bellmore,Malone, "Pathology of Travelling Salesman Subtour Elimination Algorithm" Operations Research 19, 1971, 278-307.

[2] Bodin, Golden, Assad, Ball, "The State of the Art in the Routing and Scheduling of Vehicules and Crews", 1981, Document available at National Technical Information Service, Sprinfield, Virginie, 22161.

[3] Bratley,Florian, Robillard, "Scheduling with Earliest Start and Due Date Constraints on Multiple Machines" Naval Res. Logistics Quat., vol. 22, 1975, 165-173.

[4] Christofides, Mingozzi, Toth, "State-Space Relaxation Procedures for the Computation of Bound to Routing Problems" Network, vol. 11, 1981, 145-164.

[5] Carpaneto,Toth, "Some New Branching and Bounding Criteria for the Asymmetric Traveling Salesman Problem," Management Science, 26(7), 1980, 736-743.

[6] Desrosiers, Ferland, Rousseau, Lapalme et Chapleau, "A School Busing System", Publication #164, Centre de recherche sur les transports, Montréal, 1980.

[7] Desrosiers, Pelletier, Soumis, "Plus court chemin avec contraintes d'horaires", à paraître dans RAIRO.

[8] Fisher, "Lagrangian Relaxation Method for Integer Programming Problems" Management Science, 27, 1-18, 1981.

[9] Geoffrion, "Lagrange Relaxation for Integer Programming" Math. Prog. Study 2 (197-4) 82-114.

[10] Gertsbakh, Stern, "Minimal Resources for Fixed and Variable Job Schedules" Operations Research, vol. 26 No. 1 Janv.-Feb. 1978.

[11] Levin, "Scheduling and Fleet Routing Models for Transportation Systems" Transportation Science Vol. 5, No. 3 1971.

[12] Miller, Tucker, Zemlin "Integer Programming Formulation of Travelling Salesman Problems" Journal of the Association of Computing Machinery, 7, 1960, 326-332.

[13] Orloff, "Route Constrained Fleet Scheduling" Transportation Science, Vol. 10, No. 2, 1976, 149-168.

[14] Soumis, Desrosiers, "Locomotive Maintenance Scheduling and Train Assignment" Publication #239, Centre de recherche sur les Transports, Montréal, 1983.

[15] Soumis, Desrosiers, Desrochers, "Optimal Routing and Scheduling for Public Buses with Flexibility in the Time Table", Publication #285, Centre de recherche sur les transports, Montréal, 1983.

[16] Soumis, Ferland Rousseau, "A Model for Large Scale Aircraft Routing Problems", Transportation Research B, Vol. 14, No. 1, 1980, 181-201.

[17] Swersey, Ballord, "Scheduling School Buses" Published at Yale School of Organization and Management and presented at TIMS/ORSA Conférence San-Diego 1982.